UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/698,989	10/31/2003	Cynthia A. McGuire	SUN1P763/SUN030222 NP/US	8853

62663 7590 05/30/2007

Sun Microsystems, Inc.
c/o DARBY & DARBY P.C.
P.O. BOX 770
Church Street Station
NEW YORK, NY 10008-0770

EXAMINER	
SCHELL, JOSEPH O	

ART UNIT	PAPER NUMBER
2114	

MAIL DATE	DELIVERY MODE
05/30/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/698,989	MCGUIRE ET AL.	
	Examiner	Art Unit	
	Joseph Schell	2114	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *06 March 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-61* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-9 and 11-61* is/are rejected.

7)☒ Claim(s) *10* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

Detailed Action

Claims 1-61 have been examined.

Claim 10 has been objected to as containing allowable subject matter, yet dependant upon rejected base claims.

Claims 1-9 and 11-61 have been rejected.

Response to Arguments

1. Applicant's arguments with respect to claims 26, 48, 52 and 60 have been considered but are moot in view of the new ground(s) of rejection.

Allowable Subject Matter

2. Claim 10 is objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Within the claim in its entirety, the examiner deems the novel limitation to be that each diagnostic engine subscribes to selected error reports associated with the fault diagnosis capabilities of the diagnostic engine.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

3. Claim 60 is rejected under 35 U.S.C. 101 as being directed toward non-statutory subject matter. This claim states "a fault management architecture for use in a computer system" but does not necessarily embody the architecture *in* the computer system or prevent the computer system from being an intangible signal or abstract code.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claim 61 is rejected under 35 U.S.C. 102(e) as being anticipated by Chan ('966).

Chan ('966) discloses a computer network system having a fault management architecture configured for use in a computer network system (figure 1), the computer network system comprising:

a plurality of nodes interconnected in a network (figure 1); and

a fault manager mounted at a node on the network and configured to diagnose and resolve faults occurring at said node (paragraph 26, the system may also be

employed as a stand-alone device. A stand-alone device operates to diagnose faults on itself, while it's possible existence on a network is anticipated by its embodiment in paragraph 16).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1-6, 12, 23-28, 32-34, 39-41, 45, 52-54, and 56-58 are rejected under 35 U.S.C. 103(e) as being unpatentable over Chan ('966) in view of The Many Faces of Publish/Subscribe (herein Many Faces).

6. As per claim 1, Chan ('966) discloses a fault management architecture as in claim 61, the architecture comprising:

a fault manager suitable for interfacing with diagnostic engines (Figure 7 elements 709 and 710) and fault correction agents (Figure 7, elements 715 and 717), the fault manager being suitable for receiving error information and passing this information to the diagnostic engines (as shown in Figure 7, the problem determination (element 710) diagnoses a problem (element 713) which is then passed to the error recovery logic (element 715));

at least one diagnostic engine for receiving error information and identifying a set of fault possibilities associated with errors contained in the error information (Figure 7, element 710 receives the composite log and uses rules to attempt to diagnose a problem for recovery. Also see the end of paragraph 58);

at least one fault correction agent for receiving the set of fault possibilities from the at least one diagnostic engine and then selecting a diagnosed fault, and then taking appropriate fault resolution action concerning the selected diagnosed fault (as shown in Figure 7, the problem determination module (element 710) sends problem diagnosis information (element 713) to the error recovery module (element 715) which applies the recovery process (element 717)); and

logs for tracking a status of the error information (second half of paragraph 67, the system keeps a historic record of problems and attempted solutions), a status of the fault management exercises (second half of paragraph 46, the rules are evaluated for effectiveness), and a fault status of the resources of the computer system (second half of paragraph 67, the system keeps a historic record of problems and attempted solutions).

Chan ('966) does not expressly disclose the system wherein the error information is passed to the diagnostic engines that have subscribed to receive the error information.

Many Faces discloses the use of publish/subscribe architecture for distributed applications (see abstract).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the fault management architecture disclosed by Chan ('966) such that the diagnostic engines receive data using a publish/subscribe methodology. This modification would have been obvious because it allows for the development of less cumbersome large-scale applications (Many Faces, first paragraph of the introduction).

7. As per claim 2, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 1 wherein the fault manager is configured to accommodate additional diagnostic engines and fault correction agents that can be added at a later time (Chan ('966) paragraph 44, the databases storing elements, events, known problems, conditions and recovery actions are updatable).

8. As per claim 3, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 2 wherein the fault manager is configured so that said additional diagnostic engines and additional fault correction agents can be added while the computer system is operating without interrupting its operation (Chan ('966) end of paragraph 69, new knowledge may be incorporated into the system at any time, and paragraph 49, in the exemplary rule events are continually logged. Thus the incorporation of new knowledge does not interrupt the monitoring performed by the system).

9. As per claim 4, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 1 wherein the fault correction agents resolve faults by initiating at least one of: executing a corrective action on a selected diagnosed fault (paragraph 7, courses of action to follow for treating faults are formulated, and paragraph 62, the course of action is followed) and generating a message identifying the selected diagnosed fault so that further action can be taken (Chan ('966) paragraphs 63-65, in addition to simply performing actions, the logic can send a message to cause the element to perform additional monitoring and send additional information back to the logic module).

10. As per claim 5, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 4 wherein generating a message identifying the selected diagnosed fault so that further action can be taken includes identifying faulted resource and identifying a problem with the faulted resource (in addition to sending a message to the element causing it to perform additional monitoring, the logic records to a log problem, (see Chan ('966) paragraph 67, the example in the second half of the paragraph) and its associated elements (see the beginning of Chan ('966) paragraph 58).

11. As per claim 6, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 1 wherein the architecture further includes a data capture engine configured to obtain error information from the computer system and

generate an error report that is provided to the fault manager (Chan ('966) paragraph 37, each element reports detected errors, the logic module identifies the type of the element (and thus its identity) that is reporting the error).

12. As per claim 12, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 6 wherein the diagnostic engines and the agents are configured so that the fault manager continuously accumulates error reports from the data capture engine until enough error information is accumulated so that the diagnostic engines and the agents can successfully diagnose a fault associated with the error reports (Chan ('966) paragraph 67, the determination of a degree of success of a system allows a more accurate, and thus more often successful diagnosis).

13. As per claim 23, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 1 further including a fault management statistical file that can be reviewed to determine the effectiveness of the diagnostic engines and fault correction agents at diagnosing faults and resolving faults (second half of Chan ('966) paragraph 46, the effectiveness of the rules is monitored).

14. As per claim 24, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 1 wherein the computer system comprises a single computer device (Chan ('966) paragraph 16, the system is implemented on a network of computers, thus it comprises multiple single computer devices).

15. As per claim 25, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 1 wherein the computer system comprises a plurality of computers forming a network (Chan ('966) paragraph 16).

16. As per claim 26, this claim recites limitations found in claim 1 and is rejected on the same grounds as claim 1.

17. As per claim 27, this claim recites limitations found within claim 6 and is rejected on the same grounds as claim 6.

18. As per claim 28, this claim recites limitations found within claim 12 and is rejected on the same grounds as claim 12.

19. As per claim 32, this claim recites limitations similar to claim 4, with the additional stipulation that the method be accomplished by computerized instructions. Chan ('966) in view of Many Faces discloses this additional limitation (end of Chan ('966) paragraph 36, the error recovery system may be contained within an autonomic element, and Chan ('966) paragraph 28, autonomic elements may be things like file servers and databases).

20. As per claim 33, this claim is a more general version of claim 32 and is rejected on the same grounds as claim 32. The additional possibility of software compensation for a fault is thought to be disclosed by Chan ('966) paragraph 7 (courses of action to follow for treating faults are formulated) and paragraph 62 (the course of action is followed) wherein the autonomic device being recovered is software (see paragraph 28).

21. As per claim 34, Chan ('966) in view of Many Faces discloses a method as in claim 26 wherein resolving the diagnosed fault is accomplished by implementing computerized instructions that accomplish at least one of software correction of the fault and software compensation for the fault (Chan ('966) paragraph 7 (courses of action to follow for treating faults are formulated) and Chan ('966) paragraph 62 (the course of action is followed) wherein the autonomic device being recovered is software (see paragraph 28)).

22. As per claim 39, this claim recites limitations found within claim 1 and is rejected on the same grounds as claim 1.

23. As per claim 40, this claim recites limitations found within claim 6 and is rejected on the same grounds as claim 6. The use "error report" terminology is being interpreted to mean the error event as communicated from the composite log to the problem determination logic (see end of paragraph 52)

24. As per claim 41, this claim recites limitations found within claim 12 and is rejected on the same grounds as claim 12.

25. As per claim 45, this claim recites limitations found within claim 4 and is rejected on the same grounds as claim 4.

26. As per claim 52, Chan ('966) discloses a computer network system having a fault management architecture configured for use in a computer system, the computer network system comprising:

a plurality of nodes interconnected in a network (paragraph 16);

a fault manager mounted at a first node on the network and configured to diagnose and resolve faults occurring at said first node (paragraph 26, the system may also be employed as a stand-alone device. A stand-alone device operates to diagnose faults on itself, while it's possible existence on a network is anticipated by its embodiment in paragraph 16),

the fault manager being suitable for receiving error information and passing this information to diagnostic nodes (as shown in Figure 7, the problem determination (element 710) diagnoses a problem (element 713) which is then passed to the error recovery logic (element 715)).

Chan ('966) does not expressly disclose the system wherein the error information is passed to the diagnostic engines that have subscribed to receive the error information.

Many Faces discloses the use of publish/subscribe architecture for distributed applications (see abstract).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the fault management architecture disclosed by Chan ('966) such that the diagnostic engines receive data using a publish/subscribe methodology. This modification would have been obvious because it allows for the development of less cumbersome large-scale applications (Many Faces, first paragraph of the introduction).

27. As per claim 53, with the exception of the limitations inherited from claim 52, this claim recites limitations found in claim 1 and is rejected on the same grounds as claim 1.

28. As per claim 54, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 53 wherein the fault manager is configured so that said additional diagnostic engines and additional fault correction agents can be added to the fault manager while the computer system is operating (Chan ('966) paragraph 69, the system may be modified at any time).

Chan ('966) in view of Many Faces does not explicitly disclose the system wherein the additional diagnostic engines and fault correction agents can be added without interrupting the operation of the network.

At the time of invention it would have been obvious to modify the system disclosed by Chan ('966) in view of Many Faces such that additional diagnostic engines and fault correction agents can be added without interrupting the operation of the network. This modification would have been obvious because the databases are updatable by humans (Chan ('966) paragraph 44) and the system can exist as a stand-alone system (Chan ('966) paragraph 26) or on a network (16). Thus when attached to a network, but acting as a stand-alone system (and diagnosing itself) there is no reason that a user-performed database update need interrupt the operation of the network.

29. As per claim 56, Chan ('966) in view of Many Faces discloses a computer network system having a fault management architecture as in claim 52, wherein the fault manager mounted at a first node on the network is configured to diagnose and resolve faults occurring at other nodes of the network (Chan ('966) paragraph 28, autonomic elements may be databases, web servers, etc, and paragraph 32, communication occurs over a network).

30. As per claim 57, Chan ('966) in view of Many Faces discloses a computer network system having a fault management architecture as in claim 56, wherein the

fault manager is configured to interface with diagnostic engines and fault correction agents (as shown in figure 7, error recovery (element 715) receives diagnostic information from the problem determination module (element 710) and forwards information to a recovery process (element 717), and is suitable for receiving error information and passing this information to the diagnostic engines (as shown in Figure 7, the composite log (element 709) passes information through to be diagnosed (elements 710 and 713); the fault manager including:

at least one diagnostic engine for receiving error information from the nodes of the network and diagnosing a set of fault possibilities associated with the errors contained in the error information (as shown in figure 7, elements 709 and 710, the log is analyzed to diagnose problems);

at least one fault correction agent for receiving the set of fault possibilities from the at least one diagnostic engine and then selecting a diagnosed fault from among the set of fault possibilities, and taking appropriate fault resolution action concerning the selected diagnosed fault (as shown in Figure 7, elements 710 determines a fault from multiple possibilities, and sends the data to element 715 which then causes the implementation of a recovery at element 717); and

logs for tracking a status of error information (second half of paragraph 67, the system keeps a historic record of problems and attempted solutions), a status of the fault management exercises (second half of paragraph 46, the rules are evaluated for effectiveness), and a fault status of the resources of the nodes of the network (second

half of paragraph 67, the system keeps a historic record of problems and attempted solutions).

31. As per claim 58, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 56 wherein the fault manager is configured so that said additional diagnostic engines and additional fault correction agents can be added to the fault manager while the computer system is operating (paragraph 69, the system may be modified at any time).

Chan ('966) in view of Many Faces does not explicitly disclose the system wherein the additional diagnostic engines and fault correction agents can be added without interrupting the operation of the network.

At the time of invention it would have been obvious to modify the system disclosed by Chan ('966) in view of Many Faces such that additional diagnostic engines and fault correction agents can be added without interrupting the operation of the network. This modification would have been obvious because the databases are updatable by humans (Chan ('966) paragraph 44) and the system can exist as a stand-alone system (paragraph 26) or on a network (16). Thus when attached to a network, but acting as a stand-alone system (and diagnosing itself) there is no reason that a user-performed database update need interrupt the operation of the network.

32. Claims 7-9, 29-31, 42-44 and 48 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chan ('966) in view of Many Faces and Jantz ('677).

33. As per claim 7, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 1. Chan ('966) in view of Many Faces does not explicitly disclose the fault management architecture wherein the diagnostic engine determines a probability of occurrence associated with each identified fault possibility.

Jantz ('677) teaches a system that performs error recovery according to a calculated probability of the success of each recovery procedure (see abstract and column 2 lines 59-62).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the system disclosed by Chan ('966) in view of Many Faces such that probabilities are associated with the different recovery possibilities as taught by Jantz ('677). This modification would have been obvious because a recovery procedure may be more or less useful depending on device state (Jantz ('677) column 2 lines 35-39) and it is obvious that a system that automatically implements the most probable recovery method (Jantz ('677) column 4 lines 35-37) is likely to solve a problem by implementing fewer incorrect recovery procedures than any other method of recovery decision making.

34. As per claim 8, Chan ('966) in view of Many Faces and Jantz ('677) discloses the fault management architecture of claim 7 wherein the at least one fault correction agent for receiving the set of fault possibilities receives a relative probability of occurrence associated with each identified fault possibility from the diagnostic engines (Jantz ('677) column 4 lines 6-18) and then resolves a fault using a protocol (Jantz ('677) column 4 lines 31-37, the system selects the most likely recovery procedure).

35. As per claim 9, Chan ('966) in view of Many Faces and Jantz ('677) discloses the fault management architecture of claim 8 wherein the at least one fault correction agent resolves a set of fault possibilities using a protocol that incorporates at least one of: an analysis of at least one of computer resource failure history (Chan ('966) paragraph 46, the effectiveness of new and old rules is evaluated over time to determine needed reprioritization or modification), system management policy, and relative probability of occurrence for each fault possibility (Jantz ('677) column 4 lines 31-37).

36. As per claim 29, this claim recites limitations found in claim 7 and is rejected on the same grounds as claim 7.

37. As per claim 30, Chan ('966) in view of Many Faces discloses a method as in claim 26. Chan ('966) in view of Many Faces does not expressly disclose the method wherein choosing the selected fault associated with the error information is

accomplished by implementing a computerized determination of a most likely fault associated with the error information.

Jantz ('677) teaches a system that performs error recovery according to a calculated probability of the success of each recovery procedure (see abstract and column 2 lines 59-62). The system performs a recovery by implementing the most likely recovery option (column 4 lines 30-37).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the system disclosed by Chan ('966) in view of Many Faces such that probabilities are associated with the different recovery possibilities as taught by Jantz ('677). This modification would have been obvious because a recovery procedure may be more or less useful depending on device state (Jantz ('677) column 2 lines 35-39) and it is obvious that a system that automatically implements the most probable recovery method (Jantz ('677) column 4 lines 35-37) is likely to solve a problem by implementing fewer incorrect recovery procedures than any other method of recovery decision making.

38. As per claim 31, Chan ('966) in view of Many Faces and Jantz ('677) discloses a method as in claim 30 wherein choosing the selected fault by implementing a computer determination of a most likely fault associated with the error information includes an analysis of at least one of: computer resource failure history (Jantz ('677) column 7 lines

56-61), system management policy, and relative probability of occurrence for each fault possibility.

39. As per claim 42, this claim recites limitations found in claim 9 and is rejected on the same grounds as claim 9.

40. As per claim 43, this claim recites limitations found in claim 30 and is rejected on the same grounds as claim 30.

41. As per claim 44, this claim recites limitations found in claim 31 and is rejected on the same grounds as claim 31.

42. As per claim 48, this claim recites limitations found in claim 30 and its parent claims with the additional limitations of including a processor and memory (Chan ('966) paragraph 18) and that the computer understands that the actionable fault has been resolved (Chan ('966) paragraph 46, the system monitors the effectiveness of rules by how well they solve or prevent problems, this requires knowing whether a problem has been solved by a rule).

43. Claims 11, 22, 37, and 46 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chan ('966) in view of Many Faces and Gibson (US Patent 6,343,236).

44. As per claim 11, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 1. Chan ('966) does not expressly disclose the architecture wherein the fault manager stores provided error reports in a log comprising an error report log and wherein the error report log tracks the status of the provided error reports.

Gibson ('236) teaches a system wherein faults are stored to a log that is used by the processor in assigning repairs (see abstract). The system additionally stores both unsuccessful and successful repair attempts (column 6 lines 35-39).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the system disclosed by Chan ('966) in view of Many Faces such that error status is tracked using a log. This modification would have been obvious because analyzing prior repair attempts allows the system to avoid repeatedly recommending the same repair (Gibson ('236) column 6 lines 17-19).

45. As per claim 22, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 1. Chan ('966) in view of Many Faces does not disclose the architecture further including a tool that is configured to enable a user to access the logs to determine the fault status and error history of resources in the computer system.

Gibson ('236) teaches a system that uses logs to track fault status and error history (column 1 lines 5-10 and column 4 lines 41-44, repairs are also logged, allowing for the differentiation of unresolved and resolved errors).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the error diagnosis and recovery system disclosed by Chan ('966) in view of Many Faces such that it tracks error history and status and taught by Gibson ('236). This modification would have been obvious because it allows the system to avoid recommending a previously tried repair procedure (Gibson ('236) column 6 lines 17-19) and to continually track information for outstanding un-recovered errors (Gibson ('236) column 6 lines 29-31).

At the time of invention it would have further been obvious to a person of ordinary skill in the art to modify the error diagnosis system disclosed by Chan ('966) in view of Gibson ('236) such that a user can access and view the error logs. This modification would have been obvious because the rules database is updatable by human users (Chan ('966) paragraph 44).

46. As per claim 37, Chan ('966) in view of Many Faces discloses the method as in claim 26. Chan ('966) in view of Many Faces does not explicitly disclose the method further including: providing logs for at least one of tracking errors in the system, tracking

the current status of fault diagnosis, tracking the current fault status of a resource of the computer system; and tracking a fault history of a resource of the computer system; and updating the logs based on the changes in status.

Gibson ('236) teaches a system wherein faults are stored to a log that is used by the processor in assigning repairs (see abstract). The system additionally stores both unsuccessful and successful repair attempts (column 6 lines 35-39).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the system disclosed by Chan ('966) in view of Many Faces such that error status is tracked using a log. This modification would have been obvious because analyzing prior repair attempts allows the system to avoid repeatedly recommending the same repair (Gibson ('236) column 6 lines 17-19).

47. As per claim 46, this claim recites limitations found in claim 37 and is rejected on the same grounds as claim 37.

48. Claims 13-14 and 35-36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chan ('966) in view of Many Faces and Gibson ('236), and in further view of Wikipedia's Cache Article.

49. As per claim 13, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 6 wherein the fault manager stores the error reports generated by the data capture engine to the error report log of logs (Chan ('966), second half of paragraph 67); and

wherein the at least one diagnostic engine stores fault management exercise information in a fault management exercise log of the logs (Chan ('966) second half of paragraph 46, the effectiveness of new and old rules is monitored).

Chan ('966) in view of Many Faces does not explicitly disclose the architecture wherein the fault correction agent stores fault status information concerning resources of the computer system in a resource cache of the logs.

Gibson ('236) teaches a system wherein faults are stored to a log that is used by the processor in assigning repairs (see abstract). The system additionally stores both unsuccessful and successful repair attempts (column 6 lines 35-39).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the system disclosed by Chan ('966) in view of Many Faces such that error status is tracked using a log. This modification would have been obvious because analyzing prior repair attempts allows the system to avoid repeatedly recommending the same repair (Gibson ('236) column 6 lines 17-19).

Wikipedia's Cache Article contains general information about the use of cache in a computer system.

At the time of invention it would have been obvious to a person of ordinary skill in the art to store the fault status in a cache. This modification would have been obvious because software may cache return values that are likely to be re-used in the near future (Wikipedia's Cache, first sentence under Higher-level Caches) for faster access speed (Wikipedia's Cache, first paragraph).

50. As per claim 14, Chan ('966) in view of Many Faces, Gibson ('236) and Wikipedia's Cache Article disclose the architecture of claim 13 wherein information from the error report log and the fault management exercise log are stored in the resource cache (the inclusion of the other logs in the cache would have been obvious for the same reasons as the fault status information as recited in claim 13).

51. As per claims 35-36, these claims recite limitations found in claims 13-14, respectively, and are rejected on the same grounds as claims 13 and 14.

52. Claims 15-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chan ('966) in view of Many Faces, Gibson ('236) and Wikipedia's Cache Article, and in further view of Koseki (US Patent 6,732,124).

53. As per claim 15, Chan ('966) in view of Many Faces, Gibson ('236) and Wikipedia's Cache Article discloses the fault management architecture of claim 14. Chan ('966) in view of Gibson ('236) and Wikipedia's Cache Article does not expressly disclose the architecture wherein a resource cache is configured so that in the event of a computer system failure, the system can be restarted and information can be downloaded for the resource cache to reconstruct error history, fault management exercise history, and resource status, and use this information to conduct a fault diagnosis.

Koseki ('124) teaches a system that creates a log of a storage system in secondary storage and is capable of restoring the system on an unexpected shutdown (column 2 lines 25-26 and column 7 lines 62-64).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the error diagnosis and recover system disclosed by Chan ('966) such that the system uses a secondary system for recovery in the event of a system failure. This modification would have been obvious because it allows the system to correct file storage inconsistencies caused by an unexpected crash (Koseki ('124) column 8 lines 53-61).

54. As per claim 16, Chan ('966) in view of Many Faces, Gibson ('236) and Wikipedia's Cache Article discloses the fault management architecture of claim 14.

Chan ('966) in view of Many Faces, Gibson ('236) and Wikipedia's Cache Article does not expressly disclose the architecture wherein a resource cache is configured so that in the event of a computer system failure, the system can be restarted and information can be uploaded from the resource cache to reconstruct error history, fault management exercise history, and resource status, and use this information to conduct a fault diagnosis.

Koseki ('124) teaches a system that creates a log of a storage system in secondary storage and is capable of restoring the system on an unexpected shutdown (column 2 lines 25-26 and column 7 lines 62-64).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the error diagnosis and recover system disclosed by Chan ('966) such that the system saves its logs to a secondary system for recovery in the event of a system failure. This modification would have been obvious because it allows the system to correct file storage inconsistencies caused by an unexpected crash (Koseki ('124) column 8 lines 53-61).

55. Claims 17-18, 21, 55 and 59 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chan ('966) in view of Many Faces and in further view of Coale (US Patent 4,922,491).

56. As per claim 17, Chan ('966) in view of Many Faces discloses the fault management architecture of claim 1, wherein the diagnostic engine receives an error message and diagnoses a set of fault possibilities associated with the error message (as shown in Chan ('966) Figure 7, the problem determination module (element 710) sends problem diagnoses to a recovery module (element 715)); and wherein a fault correction agent receives the set of fault possibilities from the diagnostic engines and resolves the diagnosed fault (Chan ('966) Figure 7, element 715).

Chan ('966) in view of Many Faces does not disclose the architecture including a soft error rate discriminator that: receives error information concerning correctible errors; wherein the soft error rate discriminator is configured so that when the number and frequency of correctible errors exceeds a predetermined threshold number of correctable errors over a predetermined threshold amount of time, these errors are deemed recurrent correctible errors that are sent to the diagnostic engines for further analysis.

Coale ('491) teaches a system that analyzes a peripheral, creates a problem record and creates a service alert if the problem is causing peripheral operations to degrade (see abstract). The system differentiates soft fail exceptions from hard fails, and only reacts to soft fails when it is significantly reoccurring (column 7 lines 1-9).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the error diagnosis and recovery system disclosed by Chan ('966) in view of Many Faces such that the system differentiates between hard failures and soft failures and only performs recovery of soft failures after determining that they are significantly recurring, as taught by Coale ('491). This modification would have been obvious because it allows the system to not react to random non-recurring exception events (Coale ('491) column 7 lines 9-11).

57. As per claim 18, Chan ('966) in view of Many Faces and Coale ('491) discloses the fault management architecture of claim 17 wherein the soft error rate discriminator receives error information concerning correctible errors from the diagnostic engine (Coale ('491) column 7 lines 20-25, the system uses rules to determine the differing symptoms of a soft exception and hard exception).

58. As per claims 21, 55 and 59, these claims recite limitations found in claim 17 and are rejected on the same grounds as claim 17.

59. Claims 19-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chan ('966) in view of many Faces and Coale ('491) as applied to claim 17, and in further view of Jantz ('677).

60. As per claim 19, Chan ('966) in view of Many Faces and Coale ('491) discloses the fault management architecture of claim 17. Chan ('966) in view of Many faces and Coale ('491) does not expressly disclose the architecture wherein the diagnostic engine further determines associated probabilities of occurrence for the set of fault possibilities associated with the recurrent correctible error message.

Jantz ('677) teaches a system that performs error recovery according to a calculated probability of the success of each recovery procedure (see abstract and column 2 lines 59-62). The system performs a recovery by implementing the most likely recovery option (column 4 lines 30-37).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the error diagnosis and recovery system disclosed by Chan ('966) in view of Many Faces and Coale ('491) such that recurrent soft errors are associated with recovery probabilities. This modification would have been obvious because a recovery procedure may be more or less useful depending on device state (Jantz ('677) column 2 lines 35-39) and it is obvious that a system that automatically implements the most probable recovery method (Jantz ('677) column 4 lines 35-37) is likely to solve a problem by implementing fewer incorrect recovery procedures than any other method of recovery decision making.

61. As per claim 20, Chan ('966) in view of Many Faces and Coale ('491) and in further view of Jantz ('677) discloses the fault management architecture of claim 19 wherein the a fault correction agent receives the set of fault possibilities and associated probabilities of occurrence from the diagnostic engines and the agent then takes appropriate action to resolve the set of fault possibilities (Jantz ('677) column 4 lines 32-37).

62. Claims 38 and 47 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chan ('966) in view of Many Faces and Gibson ('236) and in further view of Koseki ('124).

63. As per claim 38, Chan ('966) in view of Many Faces and Gibson ('236) discloses the method as in claim 37. Chan ('966) in view of Many Faces and Gibson ('236) does not expressly disclose the method wherein if the computer shuts down due to error, the method comprises the further steps of: restarting the system; recalling the logs tot track the fault status and fault history of resources of the computer system and thereby diagnose a fault; and resolving the fault.

Koseki ('124) teaches a system that creates a log of a storage system in secondary storage and is capable of restoring the system on an unexpected shutdown (column 2 lines 25-26 and column 7 lines 62-64).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the error diagnosis and recover system disclosed by Chan ('966) in view of Many Faces and Gibson ('236) such that the system saves logs to a secondary system for recovery in the event of a system failure. This modification would have been obvious because it allows the system to correct file storage inconsistencies caused by an unexpected crash (Koseki ('124) column 8 lines 53-61).

64. As per claim 47, this claim recites the same limitations as claim 38 and is rejected on the same grounds as claim 38.

65. Claims 49-50 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chan ('966) in view of Many Faces and Jantz ('677) as applied to claim 48, and in further view of Gibson ('236).

66. As per claim 49, Chan ('966) in view of Many Faces and Jantz ('677) discloses the computer system of claim 48. Chan ('966) in view of Many Faces and Jantz ('677) does not explicitly disclose the computer system further including instructions enabling the system to generate an error log that includes a listing of error reports.

Gibson ('236) teaches a fault logging system that generates a list of reported errors (column 1 lines 61-64).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the error correction system disclosed by Chan ('966) such that it generates an error log including a listing of error reports as taught by Gibson ('236). This modification would have been obvious because event logs typically log such error information (Chan ('966) second half of paragraph 38).

67. As per claim 50, Chan ('966) in view of Many Faces and Jantz ('677) discloses the computer system of claim 48 further including computer readable instructions enabling the computer to generate a fault management exercise log including the current status of fault diagnosis (Chan ('966) second half of paragraph 46, the effectiveness of the system rules are constantly evaluated and the probabilities adjusted).

Chan ('966) in view of Many Faces and Jantz ('677) does not explicitly disclose the system wherein the fault management exercise log that includes a listing of fault possibilities.

Gibson ('236) teaches a system wherein faults are stored to a log that is used by the processor in assigning repairs (see abstract). The system additionally stores both unsuccessful and successful repair attempts (column 6 lines 35-39).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the system disclosed by Chan ('966) such that error status is tracked using a log. This modification would have been obvious because analyzing prior repair attempts allows the system to avoid repeatedly recommending the same repair (Gibson ('236) column 6 lines 17-19).

68. Claim 51 is rejected under 35 U.S.C. 103(a) as being unpatentable over Chan ('966) in view of Many Faces and Jantz ('677) as applied to claim 48, and in further view of Gibson ('236) and Koseki ('124).

Chan ('966) in view of Many Faces and Jantz ('677) discloses the computer system of claim 48.

Chan ('966) in view of Many Faces and Jantz ('677) does not expressly disclose the computer system further including instructions enabling the computer system to generate an automatic system recovery unit log that includes a listing of the current fault status of system resources of the computer system, a listing of fault diagnosis concerning the system resources, and a listing of error reports that led to the of fault diagnosis concerning the system resource; wherein, in the event of computer system failure, upon system restart, the information in the automatic system recovery unit log can be recalled and analyzed to diagnose faults.

Gibson ('236) teaches a system wherein faults are stored to a log that is used by the processor in assigning repairs (see abstract). The system additionally stores both unsuccessful and successful repair attempts (column 6 lines 35-39).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the system disclosed by Chan ('966) such that error status is tracked using a log. This modification would have been obvious because analyzing prior repair attempts allows the system to avoid repeatedly recommending the same repair (Gibson ('236) column 6 lines 17-19).

Koseki ('124) teaches a system that creates a log of a storage system in secondary storage and is capable of restoring the system on an unexpected shutdown (column 2 lines 25-26 and column 7 lines 62-64).

At the time of invention it would have been obvious to a person of ordinary skill in the art to modify the error diagnosis and recover system disclosed by Chan ('966) in view of Gibson ('236) such that the system saves logs to a secondary system for recovery in the event of a system failure. This modification would have been obvious because it allows the system to correct file storage inconsistencies caused by an unexpected crash (Koseki ('124) column 8 lines 53-61).

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Joseph Schell whose telephone number is (571) 272-8186. The examiner can normally be reached on Monday through Friday 9AM-4:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Scott Baderman can be reached on (571) 272-3644. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

JS

SCOTT BADERMAN
SUPERVISORY PATENT EXAMINER